172nd Meeting of the Acoustical Society of America

Honolulu, Hawaii

28 November-2 December 2016

Noise: Design of Noise Control Materials: Paper 5pNSa7

Full gradient solution to adaptive hybrid control

Jacob Bean
Virginia Tech, Dpt. of Aerospace & Ocean Engineering, Blacksburg, VA; bjacob09@vt.edu

Noah H. Schiller
Structural Acoustics Branch, NASA Langley Research Center, Hampton, VA; noah.h.schiller@nasa.gov

Chris Fuller
Virginia Tech, Dpt. of Mechanical Engineering, Blacksburg, VA; cfuller@vt.edu

This paper focuses on the adaptation mechanisms in adaptive hybrid controllers. Most adaptive hybrid controllers update two filters individually according to the filtered-reference least mean squares (FxLMS) algorithm. Because this algorithm was derived for feedforward control, it does not take into account the presence of a feedback loop in the gradient calculation. This paper provides a derivation of the proper weight vector gradient for hybrid (or feedback) controllers that takes into account the presence of feedback. In this formulation, a single weight vector is updated rather than two individually. An internal model structure is assumed for the feedback part of the controller. The full gradient is equivalent to that used in the standard FxLMS algorithm with the addition of a recursive term that is a function of the modeling error. Some simulations are provided to highlight the advantages of using the full gradient in the weight vector update rather than the approximation.

1. INTRODUCTION

Active noise & vibration control (ANVC) systems composed of feedforward and feedback controllers, referred to in the literature as hybrid controllers, have been shown to offer performance benefits over solely feedforward or feedback architectures.[1,2] The performance of a feedforward control system depends on the availability of a reference signal that is correlated with the disturbance. If one is not available, or the coherence between the two is poor, the performance of the feedforward controller will deteriorate. In contrast, feedback control systems do not require a reference signal. Instead, the signal measured by the error sensor is used to generate the control signal. The performance of feedback ANVC systems is mainly influenced by the bandwidth of the disturbance and the delays associated with the plant dynamics.[3] In general, hybrid controllers are beneficial when the disturbance contains signal components that are not present in the reference signal. The feedforward controller aims to minimize the signal that is correlated with the reference signal while the feedback system simultaneously works to minimize the uncorrelated disturbance.[4]

While hybrid ANVC systems are particularly useful in certain scenarios, they have issues that need to be considered. When modeling error is present and the cost function is the mean square error (MSE), which is common in ANVC, the performance surface is no longer a convex function of the control filter coefficients. Therefore, if standard adaptive algorithms such as the filtered-reference least mean squares (FxLMS) are used to adapt the coefficients, it is possible that the global minimum might not be reached. The other main issue is that the feedback loop has the potential to cause instabilities. To remedy this, it is common to update the filter conservatively by including a leakage term in the adaptation equation,[3] which results in decreased convergence speed and degraded steady state performance. Identifying a sufficient leakage term is also, in general, a trial and error procedure.

This paper aims to alleviate the problems associated with a nonconvex performance surface such as local minima and poor convergence behavior. The ANVC algorithm considered in this paper is based on the method of steepest descent. This method seeks the minimum of a given cost function based on an estimate of the local gradient. The main contribution of the paper is a derivation of the update law for an adaptive hybrid controller employing internal model control (IMC). IMC uses an internal model of the plant to transform a feedback problem into a setting where feedforward control techniques can be applied.[5] The derivation shows that the gradient used in feedback or hybrid FxLMS is an approximation and how it can be obtained by simplifying the full gradient expression.

The derivation of the update law for a hybrid controller is based on adaptive infinite impulse response (IIR) filtering theory. The structure of the derivation is similar to adaptive IIR filtering algorithm derivations discussed by Shynk.[6] The idea is that a hybrid controller can be viewed as a single IIR filter with the feedforward part (numerator) being driven by the reference signal and the feedback part (denominator) being driven by the estimated disturbance signal. Most adaptive IIR filtering algorithms are used in system identification applications where the desired signal is known *a priori*. The equation error formulation takes advantage of this to minimize a cost function that is convex in the filter coefficients. Since in a control setting the desired signal is the external disturbance, and hence unknown, this approach cannot be used. Therefore, the output error formulation is used. This results in an error signal that is a nonlinear function of the filter coefficients.[3] The gradient of the resulting cost function, which is used in the update law, requires the computation of a recursive filter output that is a function of the modeling error and the feedback filter. A more
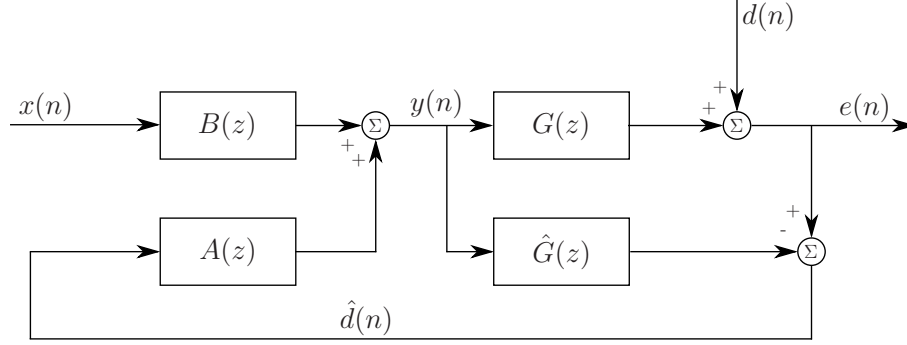
Figure 1: Block diagram of hybrid active noise control system.

computationally efficient version of the full gradient algorithm, which will be referred to as the simplified gradient algorithm, is then derived. It is then shown how FxLMS can be derived from either of these algorithms based on the assumption of zero modeling error.

A similar approach, but for adaptive feedforward control using IIR filters, was taken by Crawford and Stewart[7] Similar to this paper, they derive a full gradient and a simplified gradient algorithm following the derivation presented by Shynk.[6] The necessary simplifications to arrive at the filtered-u algorithm[8] are also discussed. In this paper, the additional term in the gradient expression is due to modeling errors in an IMC configuration whereas in Crawford and Stewart's approach,[7] the extra terms result from the presence of acoustic feedback.

The paper is organized as follows. In Section II, the adaptive algorithm is derived and the update law using the full gradient is presented. Simplifications are then made that result in a simplified gradient algorithm and the FxLMS algorithm. Section III provides simulation results to illustrate the performance differences between the simplified gradient and FxLMS algorithms. Section IV concludes the paper.

2. HYBRID CONTROL

A. FULL GRADIENT ALGORITHM

Consider the block diagram of Figure 1. This represents a standard hybrid control architecture. The plant and plant model are assumed linear time-invariant and denoted $G(z)$ and $\hat{G}(z)$, respectively. The feedforward control filter $B(z)$ is driven by the reference signal $x(n)$. The feedback controller uses an internal model of the plant to generate a reference signal $\hat{d}(n)$ that drives the feedback control filter $A(z)$. Each control filter is adapted to minimize the error signal $e(n)$. If $A(z)$ and $B(z)$ are parameterized (FIR) filters, the system can be viewed as a single adaptive IIR filter with the feedforward part being driven by the reference signal and the feedback part being driven by the estimated disturbance signal. The control signal generated by the controller can thus be written

$$y(n) = \sum_{m=1}^{N-1} a_m(n)\hat{d}(n-m) + \sum_{m=0}^{M-1} b_m(n)x(n-m), \qquad (1)$$

where $a_m(n)$ and $b_m(n)$ are the coefficients of the feedback and feedforward filters, respectively. The feedback term is summed from $m = 1$ to represent the inherent delay in estimating the distur-

bance signal. This can be written as the inner product

$$y(n) = \phi^T(n)\theta(n), \tag{2}$$

where the vector of reference signals is defined as

$$\phi(n) = \left[\hat{d}(n-1), \cdots, \hat{d}(n-N+1), x(n), \cdots, x(n-M+1)\right]^T, \tag{3}$$

and the vector of adjustable filter coefficients is

$$\theta(n) = [a_1(n), \cdots, a_{N-1}(n), b_0(n), \cdots, b_{M-1}(n)]^T. \tag{4}$$

The following adaptive algorithm, based on Widrow's LMS algorithm,[9] employs the method of steepest descent. The idea is to adapt the control filter coefficients in the negative direction of the local gradient of the cost function. The general form of the update equation takes the form

$$\theta(n+1) = \theta(n) + \alpha\left(-\nabla_{\theta(n)}J\right), \tag{5}$$

where α is the learning rate, J is the cost function, and $\nabla_{\theta(n)}$ denotes the partial derivative with respect to the current values of each filter coefficient. The cost function is the mean square error (MSE)

$$J = \frac{1}{2}E\left[e^2(n)\right]. \tag{6}$$

Because the above cost function requires the expected value of the MSE, it cannot be evaluated for online adaptation. As an approximation, it is replaced by a stochastic estimate $J(n) = \frac{1}{2}e^2(n)$. Observing that the error signal can be written as $e(n) = d(n) + G(z)y(n)$, the gradient is evaluated as

$$\begin{aligned}
\nabla_{\theta(n)}J(n) &= e(n)\nabla_{\theta(n)}\left[G(z)y(n)\right] \\
&= e(n)G(z)\nabla_{\theta(n)}\left[y(n)\right]. \tag{7}
\end{aligned}$$

The filtering by $G(z)$ can be pulled outside of the differentiation because the plant is independent of the filter coefficients. Evaluating the gradient term in (7) for a single filter coefficient gives

$$\frac{\partial y(n)}{\partial a_k(n)} = \hat{d}(n-k) + \sum_{m=1}^{N-1} a_m(n)\frac{\partial \hat{d}(n-m)}{\partial a_k(n)} \tag{8}$$

$$\frac{\partial y(n)}{\partial b_k(n)} = x(n-k) + \sum_{m=1}^{N-1} a_m(n)\frac{\partial \hat{d}(n-m)}{\partial b_k(n)}. \tag{9}$$

These equations result from applying the chain rule to (2). The second terms in (8)-(9) result from the fact that the signal driving the feedback filter, $\hat{d}(n)$ is not independent of the control filter coefficients. Computation of these derivatives is problematic because of their own recursive nature. Due to the presence of the feedback path, the past values of $\hat{d}(n)$ depend on past values of $a_k(n)$ and $b_k(n)$. With the goal of forming a proper gradient filter, it is necessary to express (8)-(9) in a form where the output is a filtered version of previous inputs and outputs. Because the partial derivatives on the right hand side are taken with respect to the current filter coefficient values, there

is no recursion. A common assumption in adaptive IIR filtering is that for slow adaptation (small α), $\theta(n) \approx \theta(n-1) \approx \cdots \approx \theta(n-N+1)$.[6] Here it is assumed that the adaptation is occurring slowly compared to the timescales (i.e., impulse response time) of the plant dynamics, which is an assumption made in the derivation of the FxLMS algorithm.[3] In this case, (8)-(9) become

$$\frac{\partial y(n)}{\partial a_k(n)} = \hat{d}(n-k) + \sum_{m=1}^{N-1} a_m(n) \frac{\partial \hat{d}(n-m)}{\partial a_k(n-m)} \tag{10}$$

$$\frac{\partial y(n)}{\partial b_k(n)} = x(n-k) + \sum_{m=1}^{N-1} a_m(n) \frac{\partial \hat{d}(n-m)}{\partial b_k(n-m)}. \tag{11}$$

To express the gradient in a true recursive form, observe that

$$\begin{aligned} \hat{d}(n-m) &= e(n-m) - \hat{G}(z)y(n-m) \\ &= d(n-m) + \Delta G(z)y(n-m), \end{aligned} \tag{12}$$

where $\Delta G(z) = G(z) - \hat{G}(z)$. Noting that $\Delta G(z)$ and $d(n-m)$ are independent of the filter coefficients, (10)-(11) become

$$\frac{\partial y(n)}{\partial a_k(n)} = \hat{d}(n-k) + \Delta G(z) \sum_{m=1}^{N-1} a_m(n) \frac{\partial y(n-m)}{\partial a_k(n-m)} \tag{13}$$

$$\frac{\partial y(n)}{\partial b_k(n)} = x(n-k) + \Delta G(z) \sum_{m=1}^{N-1} a_m(n) \frac{\partial y(n-m)}{\partial b_k(n-m)}, \tag{14}$$

which are recursive in the partial derivatives. For convenience, define

$$u_k(n) = \frac{\partial y(n)}{\partial a_k(n)}, \qquad v_k(n) = \frac{\partial y(n)}{\partial b_k(n)}, \tag{15}$$

so that (13)-(14) can be written

$$u_k(n) = \hat{d}(n-k) + \Delta G(z) \sum_{m=1}^{N-1} a_m(n) u_k(n-m) \tag{16}$$

$$v_k(n) = x(n-k) + \Delta G(z) \sum_{m=1}^{N-1} a_m(n) v_k(n-m). \tag{17}$$

Equations (16)-(17) are expressed in form of recursive filters as

$$u_k(n) = \left(\frac{1}{1 - \Delta G(z)A(z)} \right) \hat{d}(n-k) \tag{18}$$

$$v_k(n) = \left(\frac{1}{1 - \Delta G(z)A(z)} \right) x(n-k), \tag{19}$$

where

$$A(z) = a_1(n)z^{-1} + \cdots + a_{N-1}(n)z^{-N+1}. \tag{20}$$

Note that there are no restrictions on the structure of $\Delta G(z)$; it may be parameterized as a scalar, an FIR, or an IIR filter.

It is helpful to form a vector of filtered reference signals. This is done by filtering each element $u_k(n)$ and $v_k(n)$ through the plant model. This is written

$$u_{k_f}(n) = \hat{G}(z)u_k(n) \quad k = 1, \cdots, N-1 \tag{21}$$

$$v_{k_f}(n) = \hat{G}(z)v_k(n) \quad k = 0, \cdots, M-1. \tag{22}$$

The vector of filtered reference signals is then written as

$$\phi_f(n) = \left[u_{1_f}(n), \cdots, u_{N-1_f}(n), v_{0_f}(n), \cdots, v_{M-1_f}(n) \right]. \tag{23}$$

The gradient in (7) can now be written as

$$\nabla_{\theta(n)} J(n) = e(n)\phi_f(n). \tag{24}$$

The resulting update equation is

$$\theta(n+1) = \theta(n) - \alpha\phi_f(n)e(n). \tag{25}$$

The filtered reference signals are generated using the effective plant response, which takes into account the presence of the feedback path. Taking the feedback path into account directly in the adaptation equation allows the algorithm to converge faster than gradient descent algorithms that ignore it such as feedback and hybrid FxLMS. The only additional computations necessary to calculate the proper gradient involve the recursive filtering by $1 - \Delta G(z)A(z)$.

It is important to differentiate between adaptive algorithm divergence and feedback loop instabilities.[10] The inclusion of the recursive term in the gradient calculation does not prevent the feedback path from going unstable if the loop gain is greater than unity at $180°$ crossovers. Instead, it essentially gives the adaptive algorithm a more accurate estimate of the performance surface of which it is navigating. If the filter coefficients are adapted into a region that destabilizes the feedback loop, the controller becomes unstable and the MSE will diverge.

While the formulation thus far has been for a hybrid control structure, it should be emphasized that this is also applicable to feedback control. As long as the signal driving the adaptive filter is a function of the adaptive filter itself, this approach can be taken.

B. SIMPLIFIED GRADIENT ALGORITHM

The summation terms (16)-(17) are each filtered by the modeling error, which requires computing the output of $2(M+N-1)$ filters. The full expressions for $u_k(n)$ and $v_k(n)$ are then filtered through the plant model as in (21)-(22), which requires computing the output of $M+N-1$ more filters. The full gradient algorithm thus requires computation of $3(M+N-1)$ filter outputs, which can be cumbersome if the control filters contain a large number of coefficients. Rather than computing each filter output, it is possible to calculate only the initial gradient terms ($u_{1_f}(n)$ and $v_{0_f}(n)$) and approximate the remaining terms as delayed versions of the initial gradients. This is another common simplifying assumption in adaptive IIR filtering that results in negligible performance loss[6] in most situations. The initial gradient terms are calculated as

$$u_1(n) = \hat{d}(n-1) + \Delta G(z)\sum_{m=1}^{N-1} a_m(n)u_1(n-m) \tag{26}$$

$$v_0(n) = x(n) + \Delta G(z) \sum_{m=1}^{N-1} a_m(n) v_0(n-m). \tag{27}$$

Each of these terms are then filtered through the plant model

$$u_{1_f}(n) = \hat{G}(z) u_1(n) \tag{28}$$

$$v_{0_f}(n) = \hat{G}(z) v_0(n). \tag{29}$$

The remaining terms are approximated as

$$u_{k_f}(n) = u_{1_f}(n-k) \qquad k = 2, \cdots, N-1 \tag{30}$$

$$v_{k_f}(n) = v_{0_f}(n-k) \qquad k = 1, \cdots, M-1. \tag{31}$$

With this simplification, the outputs of only six filters are computed to form the gradients at each iteration. This approximation is valid under the previously made assumption of slow adaptation.

C.   FXLMS ALGORITHM

The FxLMS algorithm can be derived quite simply from the formulation presented above. If a perfect plant model is assumed, then $\Delta G(z) = 0$. Then the gradient is simply delayed versions of the reference signals such that

$$u_k(n) = \hat{d}(n-k) \tag{32}$$

$$v_k(n) = x(n-k). \tag{33}$$

The filtered reference signals used in the algorithm are these signals filtered through the plant model. In vector form, this is

$$\phi_f(n) = \hat{G}(z) \phi(n). \tag{34}$$

The update equation for the FxLMS algorithm is then

$$\theta(n+1) = \theta(n) - \alpha \phi_f(n) e(n). \tag{35}$$

This is equivalent to the standard formulation of the FxLMS algorithm for a hybrid controller, which updates the feedback and feedforward control filters individually.

The feedback FxLMS algorithm can be viewed as analogous to the pseudolinear regression (PLR) algorithm in adaptive IIR filtering.[11] Both algorithms neglect the recursive filtering in the calculation of the gradient. The PLR algorithm has a self-stabilizing property such that when system poles migrate outside of the unit circle the adaptation naturally forces them back inside.[6] This behavior can also be seen in adaptive feedback and hybrid systems employing the FxLMS algorithm.

D.   QUANTIFICATION OF MODELING ERROR

The complete gradient expression ((18)-(19)) has been shown to be a function of the plant modeling error $\Delta G(z)$. The question of how to characterize this quantity then naturally arises. As was shown in the previous section, if no modeling error exists (never the case in practice), the full gradient algorithm reduces to FxLMS. To calculate the precise instantaneous gradient in (21)-(22),

it would be necessary to have exact knowledge of the modeling error. If the exact modeling error was known, it would make more sense to use that knowledge to generate a more accurate plant model and then use the simpler FxLMS algorithm to adapt the control filter. A practical solution is to design a $\Delta G(z)$ that offers performance at least comparable (and ideally superior) to that of FxLMS for a family of possible plants.

3. SIMULATION

It has been shown that the update equation in the feedback and hybrid FxLMS algorithm uses an approximation of the full gradient expression. A series of simulations have been constructed to show the performance differences in using the complete gradient as opposed to the approximation used in FxLMS. To save on computing time, the simplified gradient algorithm ((26)-(31)) is used in all simulations. The simplification results in almost no loss of performance.

In each simulation, the reference signal is a 100 Hz sine wave. The disturbance is the reference signal filtered through the primary path plus a 140 Hz sine wave (i.e., an uncorrelated disturbance). The primary path from the reference signal to the error sensor is a five sample delay with unity magnitude. A low level broadband signal was added to the disturbance to simulate measurement noise. In all simulations, the sample rate was set to 1 kHz. The plant model is represented by the following FIR filter

$$
\begin{aligned}
\hat{G}(z) = \quad & - \quad 0.03 + 0.3z^{-1} + 1.4z^{-2} + 0.9z^{-3} - 0.4z^{-4} \\
& - \quad 1.1z^{-5} - 0.2z^{-6} + 0.3z^{-7} + 0.07z^{-8},
\end{aligned} \tag{36}
$$

which is representative of a simple ANVC plant. It is assumed that the plant is subject to a multiplicative uncertainty $\Delta_m(j\omega)$, where

$$
G(j\omega) = \hat{G}(j\omega)\left(1 + \Delta_m(j\omega)\right) \tag{37}
$$

and that the uncertainty is bounded by $|\Delta_m(j\omega)| \leq \frac{1}{8}$. In other words, the magnitude of the plant is expected to fall in the range

$$
\frac{7}{8}\hat{G}(j\omega) \leq G(j\omega) \leq \frac{9}{8}\hat{G}(j\omega). \tag{38}
$$

Therefore, for the simulations, it was assumed that the modeling error $\Delta G(j\omega)$ had a magnitude of $|\frac{1}{8}\hat{G}(j\omega)|$ and zero phase. The MATLAB function *invfreqz()* was then used to generate a corresponding FIR filter that approximated the desired frequency response. This FIR filter is used in the simplified gradient algorithm in all subsequent simulations.

To compare the performance of the simplified gradient algorithm and FxLMS, three separate simulations are presented. In the first two cases, $G(j\omega) = \frac{3}{4}\hat{G}(j\omega)$ and $G(j\omega) = \frac{5}{4}\hat{G}(j\omega)$. In these cases, the multiplicative error is $\frac{1}{4}$ and $-\frac{1}{4}$, respectively. Both cases exceed the bounds set in (38). In the final simulation, $G(z) = \hat{G}(z)$.

The control filter consists of two feedback ($N = 3$) and two feedforward ($M = 2$) coefficients. In general, only two filter weights should be necessary for the suppression of a tone. In each simulation, the step size was adjusted such that the fastest convergence for each algorithm was achieved. Any further increase caused instability or oscillatory behavior in the MSE signal that

led to a longer convergence time. The results are shown in Figs. (2)-(4). For both cases where the plant model is imperfect, the simplified gradient algorithm converges faster than FxLMS due to a more accurate gradient estimate in the update equation. In the nominal case, the system is entirely feedforward and the gradient used in the FxLMS update is correct. Therefore, it is expected that FxLMS will converge faster than the simplified gradient algorithm. This is confirmed by the results in Fig. 4.
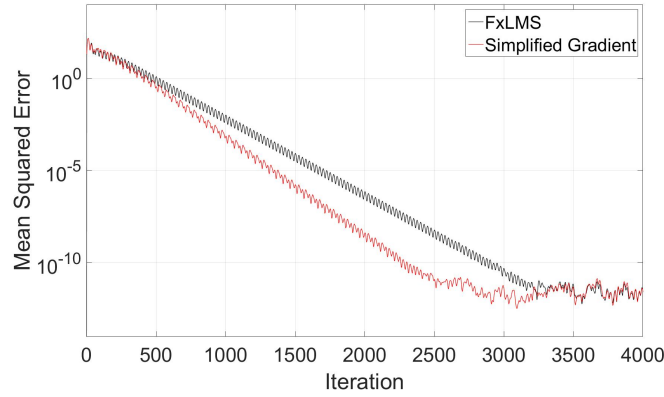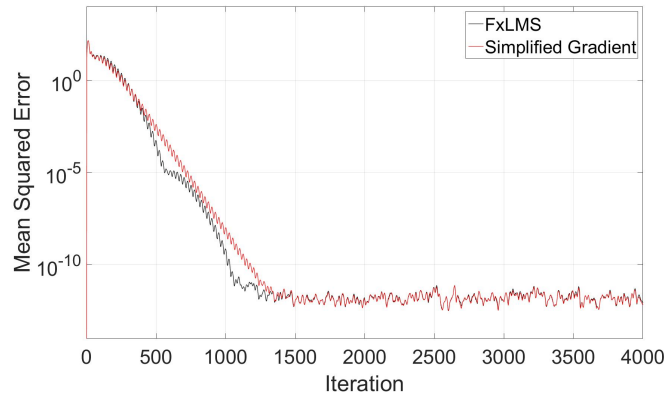

Figure 2: Mean squared error signal in the case of modeling errors: $G(z) = \frac{3}{4}\hat{G}(z)$.
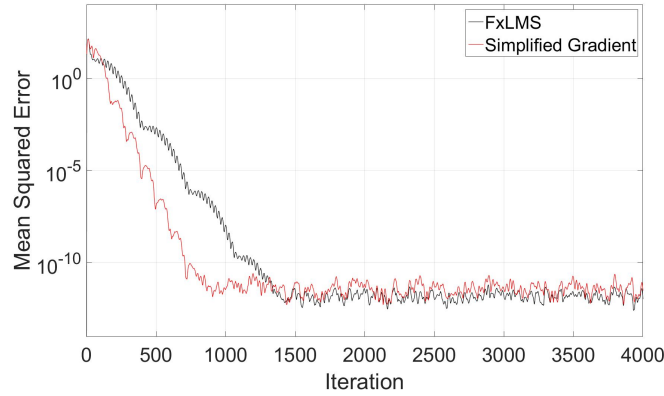
Figure 3: Mean squared error signal in the case of modeling errors: $G(z) = \frac{5}{4}\hat{G}(z)$.

4. CONCLUSIONS

A solution to adaptive internal model control based on the method of steepest descent has been derived that makes no upfront assumptions regarding the accuracy of the plant model. The only assumption made was slow control filter adaptation. This algorithm was then simplified into a more computationally efficient form. The update law employs a gradient that was shown to be a function of the plant modeling error. From the existing formulation, the FxLMS algorithm was then derived

Figure 4: Mean squared error signal in the case of no modeling errors: $G(z) = \hat{G}(z)$.

based on the assumption of zero modeling error. Simulation results show that the complete update law is capable of converging faster than FxLMS when an estimate of the plant modeling error is available. In practice, we have found that the FxLMS algorithm provides sufficient performance in many situations. Nonetheless, this approach to the derivation of FxLMS provides helpful insight into the algorithm's behavior when applied in a hybrid or feedback control setting.

REFERENCES

[1] Kong, Xuam; Liu, Pu; Kuo, Sen M. "Multiple channel hybrid active noise control systems." *IEEE transactions on control systems technology* 6 (6), (1998): 719-729.

[2] Ray, Laura R.; Solbeck, Jason A.; Streeter, Alexander D.; Collier, Robert D. "Hybrid feedforward-feedback active noise reduction for hearing protection and communication." *The Journal of the Acoustical Society of America* 120 (4), (2006): 2026-2036.

[3] Elliott, Stephen. Signal processing for active control. Academic press, 2000.

[4] Akhtar, Muhammad T.; Mitsuhashi, Wataru. "Improving performance of hybrid active noise control systems for uncorrelated narrowband disturbances." *IEEE transactions on audio, speech, and language processing* 19 (7), (2011): 2058-2066.

[5] Morari, Manfred; Zafiriou, Evanghelos. Robust process control. Englewood Cliffs, NJ: Prentice hall, 1989.

[6] Shynk, John J. "Adaptive IIR filtering." *IEEE Assp Magazine* 6 (2), (1989): 4-21.

[7] Crawford, David H.; Stewart, Robert W. "Adaptive IIR filtered-v algorithms for active noise control." *The Journal of the Acoustical Society of America* 101 (4), (1997): 2097-2103.

[8] Eriksson, L. J. "Development of the filtered-U algorithm for active noise control". *The Journal of the Acoustical Society of America* 89 (1), (1991): 257-265.

[9] Widrow, Bernard; Stearns, Samuel. Adaptive Signal Processing. Prentice Hall, 1985.

[10] Vaudrey, Michael A.; Baumann, William T.; Saunders, William R. "Stability and operating constraints of adaptive LMS-based feedback control." *Automatica* 39 (4), (2003): 595-605.

[11] Feintuch, Paul L. "An adaptive recursive LMS filter." *Proceedings of the IEEE* 64 (11), (1976): 1622-1624.